ControlPay: An Adaptive Payment Controller for Blockchain Economies

Oguzhan Akcin, Robert P. Streit, Benjamin Oommen, Sriram Vishwanath, and Sandeep Chinchali
The University of Texas at Austin
{oguzhanakcin, rpstreit, baoommen, sriram, sandeepc}@utexas.edu

Abstract*—Recent advancements in blockchain technology have led to the development of various decentralized service platforms for various tasks, like machine learning and wireless networks for example. Central to the operation of these platforms is a token-based economy, rewarding service providers with cryptocurrency tokens for their contributions to the setup, verification, and maintenance of a platform. However, these platforms often rely on predetermined token supply strategies which render a platform's operation susceptible to market fluctuations. A more flexible approach, one allowing for dynamic response to changes in system demand and market conditions, is essential to mitigate such vulnerabilities. To address these challenges, we introduce a control-theoretic approach to stabilizing a decentralized service platform's token economy. Specifically, we first model these blockchain economies as dynamical systems where token circulation, pricing, and consumer demand evolve based on payments to service providers and service costs. Then, we utilize our model to introduce ControlPay: a novel payment controller based on model predictive control (MPC) designed to enhance the performance of decentralized networks while simultaneously ensuring token price stability. Additionally, we also examine the impact of strategic behavior in the market through a Stackelberg game to further enhance the robustness of our payment controller. Finally, we evaluate our methodology on real and synthetic data. Our findings show that ControlPay significantly outperforms conventional algorithmic stablecoin approaches, yielding improvements of up to $2.4\times$ in simulations based on actual demand data from existing blockchain-driven decentralized wireless networks.

I. Introduction

The functionality of blockchain based service platforms is rapidly changing demand economies for machine learning, decentralized wireless communications, storage, delegated computation, and electric vehicle charging networks, among other areas. For example, Helium [1] and Pollen [2] are two prominent decentralized wireless networks (DeWi) that reward independent service providers to build, maintain, validate, secure, and ultimately send data over 5G hotspots in a distributed manner. Similarly, projects such as BitTensor [3], FileCoin [4], Storj [5], and ComputeCoin [6] offer decentralized machine learning, file storage, and computing services. Underlying their operation, these networks reward suppliers using a corresponding token (cryptocurrency) to maintain and offer services over a decentralized platform. Likewise, consumers can exchange US dollars (USD) for tokens enabling the utilization of services or participation in the associated crypto-economy if they so choose.

To regulate the token rewards to service providers, several projects have recently considered adopting a "burn-and-mint"

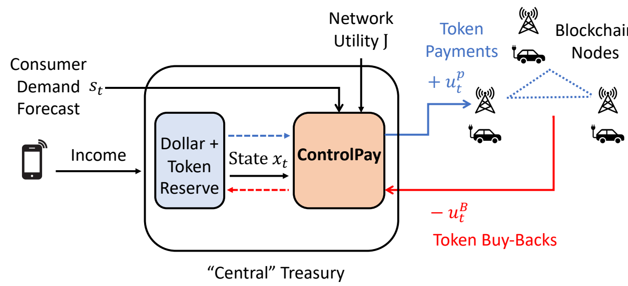

Fig. 1: **A Control System for Blockchain Tokenomics.** We design a payment controller (orange), ControlPay, to optimize the network utility J. ControlPay takes in a forecast of consumer demand and supplier growth s_t, as well as the treasury state x_t. Then, it adaptively controls token payments u_t^P and buy-backs u_t^B to achieve a stable token price.

token economics (tokenomics) model where a central reserve "mints" tokens to reward suppliers while, correspondingly, tokens are "burnt" (deleted from the circulating supply) when consumers want to use network services. By burning tokens, the system reduces the token supply to keep the inflation in token prices under control. However, current approaches try to maintain a hard peg, i.e. a fixed exchange rate between cryptocurrencies and the USD, which may result in the depletion of cryptocurrency and dollar reserves of the system as the demand for cryptocurrency fluctuates. An ideal token economy must be designed with adaptable mechanisms for regulating token rewards to achieve stable token prices while preserving the token and dollar reserves. As such, the number of tokens should gracefully scale with the size of the infrastructure network, which one cannot necessarily know a priori.

When treated as a mechanism design problem, one identifies that burn-and-mint strategies will induce market equilibria which are colloquially referred to as burn-and-mint equilibrium (BME) [7]. We posit that a BME must be "programmable" so that blockchain-based service platforms can maximize the total utility of all users. For example, a network utility function (performance criterion) can be chosen to include maintaining a steadily growing token price with low volatility. Likewise, a designer can choose a utility function incentivizing new suppliers/consumers to expand geographical coverage. Moreover, the BME-based token economy could be designed to satisfy strict performance guarantees and constraints, such as limiting the number of tokens minted and

2834-9946/24/$31.00 ©2024 IEEE
DOI 10.1109/Blockchain62396.2024.00048

burned per day. Taking this even further, participants in the token economy are likely rational so it is important to consider their agency, and any impacts, in taking actions to maximize the value of their holdings. In short, there is a need for solutions deployed in infrastructure-centric blockchain networks to address these aspects of managing token supply.

Our fundamental observation is that token economies can be modeled as dynamical systems, allowing us to leverage **control theory** to maximize the blockchain network's utility function under chosen constraints. Control theory is a natural tool since the token economy is a dynamical system where the circulating token supply, token price, and token reserves change as a function of our burn and mint decisions. Furthermore, a decentralized consensus mechanism has the authority to control token regulations at the protocol level transparently. And, by stating a system's desiderata, one can identify a control cost function capturing key metrics supporting the long-term performance and evolution of the blockchain system. Hence, regulating a blockchain token economy is a model-based control problem, solvable via optimal control theory.

Our Contributions: Overall, the contributions of this paper are three-fold. **(1)** To the best of our knowledge, we are the first to apply optimal control theory to blockchain tokenomics and introduce a general-purpose dynamical systems model that flexibly captures fixed-supply systems, burn-and-mint systems, and various other network cost functions. **(2)** We design ControlPay, a novel payment controller for a token economy inspired by nonlinear model predictive control (MPC) and game-theoretic methods that are used in high-performance, safety-critical applications like autonomous driving [8, 9], machine learning [10], and rocket guidance [11]. We demonstrate that these methods perform better than common heuristic controllers, such as proportional integral derivative (PID) controllers used by some algorithmic stablecoins. Specifically, we improve on PID by $2.4\times$ on simulated time series demand patterns and by $2.7\times$ on real demand patterns from the Helium DeWi Blockchain. **(3)** Finally, to maximize network welfare we introduce a novel game theoretic formulation for how owners of tokens and our controller strategically interact.

II. Related Work

Prior studies on blockchains as dynamical systems [12–14] primarily focus on miner profitability and how block rewards influence supply and demand. Unlike these works, our work centers on infrastructure-centric blockchain systems, where supply and demand dynamics are governed by controller actions and predictive models. Thus, our controller specification is decoupled from the possibly complex trajectory of the demand, and the strength of our controller's predictions relates to the strength of the forecasts used in the system.

To better understand the robustness of our methodology, we also consider the impact of rational behavior on the part of consumers in our system. Game theoretic analyses of blockchain systems have a long tradition, starting with the original Bitcoin white paper [15]. Since the discovery of the selfish mining attack [16], game theoretic methods have been used to investigate rational deviations [17], mining pools [18], and more recently transaction fee auctions in Ethereum-like blockchains [19, 20]. Our work differs from existing literature as we focus on the effects of rational behavior on buy-back and pay strategies used to stabilize token prices, and not on modeling its effects on any underlying blockchain protocol.

Before continuing, we briefly mention prior investigations into the application of control theory to monetary policy. For instance, some authors have exploited the linear system dynamics of central banking to use interest rates for optimal control of a currency's inflation rate [21, 22]. Our work is not directly comparable due to the difference in application. Namely, we are concerned with a control system that follows a pre-specified price trajectory of a token in a Blockchain system, and to do so we exploit non-linear supply and demand dynamics within the token economy. It is also interesting to note that researchers have argued empirically on real-world data [23], as well as in the abstract on simulated environments [24], that current monetary policies can be captured by PID controller methods. Such approaches are heuristic and reactive (instead of predictive), and our experiments in Section VI will show that our MPC-based approach outperforms them.

Finally, as our aim is to stabilize a token price in a blockchain network, our work bears similarity to algorithmic stablecoins. However, our interests are in adaptively controlling the circulating supply of a token to balance payments to service providers with a pre-specified control trajectory on the token price. Thus, our work is more related to service networks employing burn-and-mint systems like Helium [1] and Factom [25] than more general-purpose stable-coins like Reflexer [26] or Terra [27]. Furthermore, most of the existing literature is *reactive* through the use of heuristic methods, whereas our work is *predictive* through optimal adaptive control methods. As our focus is on infrastructure networks, our work is applicable to DeWi [28] scenarios like Helium [1], as well as file sharing [4] and decentralized video streaming [29].

III. The Token Economy as a Dynamical System

We now model the token economy for infrastructure networks as a dynamical system to capture the following: *Nodes* provide services (5G base stations, machine learning inference, etc.) to *consumers*, who pay the controller in dollars. The controller, in turn, rewards nodes with tokens, a process termed as *minting*. The income from consumers forms a dollar reserve, parts of which are used to buy back tokens, thereby reducing the circulating supply – a process referred to as *burn* action. We define the system dynamics and variables in what follows.

Circulating Supply: The circulating supply of tokens S_t increases when u_t^{P} tokens are paid as rewards to nodes and decreases when tokens are bought back:

$$S_{t+1} = S_t + \underbrace{u_t^{\mathrm{P}}}_{\text{Tokens Paid}} - \underbrace{\frac{u_t^{\mathrm{B}}}{p_t^{\mathrm{Tok}} + \Delta p_t}}_{\text{Tokens Bought Back}} . \quad (1)$$

Here, u_t^{B} is the number of dollars the controller pays to token owners to purchase their tokens at the price of $(p_t^{\mathrm{Tok}} + \Delta p_t)$.

We define p_t^{Tok} as the current market price of the token, while Δp_t is the extra amount the controller pays the users over the market price to incentivize them to part with their tokens. We formulate a mathematical game to calculate Δp_t for rational agents in Section V.

Reserve: The controller holds two reserves, one comprising tokens R_t^{Tok} and the other dollars R_t^{USD} with quantities varying over time steps t. The dollar reserve increases with the income received from users purchasing tokens and decreases with buybacks from the market while the token reserve R_t^{Tok} increases with buybacks and decreases with payments made to the service providers:

$$R_{t+1}^{\text{USD}} = R_t^{\text{USD}} - u_t^{\text{B}} + \text{Inc}_t, \tag{2}$$

where u_t^{B} is the dollars used to buy back tokens at time t, and Inc_t denotes the income received from users purchasing tokens. Analogous to the dollar reserve, the token reserve changes as:

$$R_{t+1}^{\text{Tok}} = R_t^{\text{Tok}} - u_t^{\text{P}} + \frac{u_t^{\text{B}}}{p_t^{\text{Tok}} + \Delta p_t}. \tag{3}$$

Token Price: The token price at time t is given by p_t^{Tok}, and we assume that it is market clearing. Thus, for circulating supply S_t and demand for tokens D_t at time t, we have:

$$p_t^{\text{Tok}} = \frac{D_t}{S_t}.$$

Here, demand D_t can be forecasted using historical data, as shown for public Helium DeWi data in Section VI.

State and Control Variables: The state x_t captures the dynamic quantities that are necessary to control the system. Likewise, the control vector u_t consists of how much we adaptively pay, buyback, and our incentive price:

$$x_t = \begin{bmatrix} S_t, & R_t^{\text{USD}}, & R_t^{\text{Tok}}, & p_t^{\text{Tok}} \end{bmatrix}^\top, u_t = \begin{bmatrix} u_t^{\text{B}}, & u_t^{\text{P}}, & \Delta p_t \end{bmatrix}^\top.$$

Additionally, as our method is *predictive*, we use forecasts that predict future income and consumer demand. In practice, these can come from data-driven modeling using historical transaction data. The forecasts at time t are:

$$s_t = \begin{bmatrix} \widehat{D_t}, & \widehat{\text{Inc}_t} \end{bmatrix}^\top,$$

where $\widehat{D_t}$ and $\widehat{\text{Inc}_t}$ correspond to estimates of demand and income, respectively. Now, recalling the specification of the state vector x_t above, we write the dynamics of our system with Eq. 1–3:

$$x_{t+1} = f(x_t, u_t, s_t) = \begin{bmatrix} x_t(0) + u_t(1) - \frac{u_t(0)}{x_t(3)+u_t(2)} \\ x_t(1) + s_t(1) - u_t(0) \\ x_t(2) + \frac{u_t(0)}{x_t(3)+u_t(2)} - u_t(1) \\ \frac{s_{t+1}(0)}{x_{t+1}(0)} \end{bmatrix}. \tag{4}$$

State/Control Constraints: Now we define the feasible state set $\mathcal{X}$ and control set $\mathcal{U}$. The state x_t should be non-negative and above a safety margin since it encodes the circulating supply, token dollar reserves, etc. For example, we might want the dollar reserve above a positive safety margin, specifically $R_t^{\text{USD}} \geq R_t^{\text{USD,min}}$. Likewise, we want to pay and buy back a positive number of tokens, given by $u_t^{\text{P}}, u_t^{\text{B}} > 0$. There are no constraints on Δp_t, as tokens can be offered for buyback below market price without any guarantees of sale success.

Network Cost Function: The network cost function J encodes the token economy's high level performance criterion. To make this precise, suppose we want the price to follow a smooth and increasing reference trajectory, denoted by $\bar{p}_t^{\text{Tok, ref.}}$. As our goal is to control the supply to match this trajectory, a natural objective function is to minimize the difference in L_2-norm between the observed price p_t^{Tok} and the reference price $\bar{p}_t^{\text{Tok, ref.}}$. Moreover, one may penalize the difference between the enacted payments u_t^{P} and buybacks u_t^{B} from a reference payment regime $\bar{u}_t$ to minimize the intervention to the economy. Putting it all together, we aim to minimize the following network cost function J:

$$J(x_0, u_{0:H-1}) = \sum_{t \in \mathcal{T}} \begin{bmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \end{bmatrix}^\top \begin{bmatrix} (x_{t+1}(3) - \bar{x}_{t+1}(3))^2 \\ (u_t(0) - \bar{u}_t(0))^2 \\ (u_t(1) - \bar{u}_t(1))^2 \end{bmatrix}, \tag{5}$$

where $\mathcal{T} = \{0, \dots H-1\}$ is the set of time steps over the horizon H. The first entry of the right vector is the L_2-error in following the price reference trajectory, while the second and third correspond to penalties in L_2-norm on the chosen buyback and reward controls, respectively, against a reference $\bar{u}_t$. The parameters $\beta_1, \beta_2, \beta_3$ are a design choice to trade off how closely the reference price is tracked and the control effort. We note here that one could also adopt any differentiable, nonconvex cost function amenable to gradient-based optimization.

IV. Payment Controller

We now present our proposed payment controller, ControlPay, to control the token economy described in Section III. Our controller is a model predictive control (MPC) based approach that starts by acquiring a stochastic forecast of income, denoted as s_t, reflecting consumer demand. Then, given the current state x_0 and forecast $\hat{s}_{0:H-1}$, we can propagate the known dynamics $f(x_t, u_t, s_t)$ according to Eq. 4. Since we have forecasts of node and consumer growth for H steps in the future, we naturally have a finite horizon control problem of H steps. Our goal is to optimize the performance metric, which is to minimize the aggregate network cost J. More formally, our ControlPay solves the following problem in each time step to obtain the optimal sequence of controls $u_{0:H-1}^*$:

$$\begin{aligned} \underset{u_{0:H-1}}{\text{minimize}} \quad & J(x_0, u_{0:H-1}), \\ \text{subject to} \quad & x_{t+1} = f(x_t, u_t, s_t), && \forall t \in \mathcal{T}, \\ & x_{t+1} \in \mathcal{X}, && \forall t \in \mathcal{T}, \\ & u_t \in \mathcal{U}, && \forall t \in \mathcal{T}. \end{aligned} \tag{6}$$

After obtaining the optimal sequence of controls $u_{0:H-1}^*$, we implement the first control u_0^*, observe the next state x_1, and re-plan based on the observation. This re-planning helps mitigate forecast uncertainty in the system [30]. In order

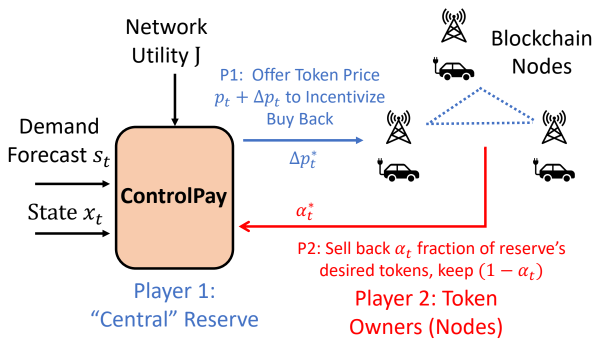

Fig. 2: **Strategic Pricing as a Game.** The reserve (player 1) offers to buy back tokens at the market price of p_t^{Tok} plus a bonus incentive of Δp_t to reduce the circulating supply. However, all token-owning nodes (collectively, player 2) might want to sell only a fraction α_t of tokens since they could be valuable in the future. We solve for an optimal game strategy for each player (Δp_t^* and α_t^*) using bi-level optimization.

to solve Eq. 6, non-convex optimization methods such as Sequential Convex Programming (SCP) [31, 32] can be used.

V. Strategic Pricing: A Game Theoretic Analysis

In our control theoretic formulation, we assume that token owners will sell their tokens to the reserve when it offers to buy back tokens with an incentive price of Δp_t. However, as shown in Fig. 2, strategic token owners might only sell a fraction of their tokens for immediate revenue and retain the rest for their future expected value. As such, ControlPay must offer a sufficiently high incentive Δp_t to token owners to sell their tokens so that the circulating token supply is regulated to avoid inflation. Our key insight is that strategic pricing can be formulated as a two-player Stackelberg game (see [33]).

Market Dynamics: The market dynamics arise from the selfish behavior of rational consumers. At each time step, we have a sequential game between the reserve's controller (player 1) and all token-owning nodes (player 2). The controller optimizes the program (6) and the consumers seek to maximize the value of their token holdings over the time horizon. For simplicity, we assume this interaction is a game of complete information. For example, token owners are aware of the controller's strategy, which is encoded in smart contracts distributed across the blockchain. Likewise, each player also perfectly observes the token price and supply.

The controller moves first by posting a price $(p_t^{\text{Tok}} + \Delta p_t)$. Then, the consumers respond by selling an α_t fraction of their collective holdings, where α_t is chosen to maximize the value of the consumers' holdings (utility U_t) at time step t. In each time step, agents are also planning over the time horizon H, thus the decision variable is a vector $\boldsymbol{\alpha}_t = [\alpha_t, \alpha_{t+1}, \cdots, \alpha_{t+H}]^\top$ such that $\mathbf{1}^\top \boldsymbol{\alpha_t} = 1$ and $\boldsymbol{\alpha_t} \geq \mathbf{0}$. We will denote $\boldsymbol{\alpha_t}$ for a vector and α_t for scalars.

The utility U_t for a token-owner can be any concave function over the time horizon H. A token-owner's utility at time t is a function of present prices, their beliefs about future prices, their holdings, and the supply. A natural formulation for the utility U_t of the token-owners is the sum of the earnings received today and the earnings they speculate can be earned in the future over the time horizon H. We describe this utility as a function of the game strategies as well as other parameters of the environment:

$$U_t = \underbrace{\alpha_t \cdot S_t(p_t^{\text{Tok}} + \Delta p_t)}_{\substack{\text{Current earnings from selling} \\ \alpha_t\text{-fraction of supply.}}} + \underbrace{\sum_{j=1}^{H} \alpha_{t+j} \gamma^j S_t \mathbb{E}(p_{t+j}^{\text{Tok}})}_{\substack{\text{Future expected earnings from} \\ \text{holding } 1-\alpha_t\text{-fraction of supply.}}} \; . \quad (7)$$

Here, γ is a discount factor that attenuates the expected future earnings from not selling in the current time step. Additionally, the randomness in the expectation is due to forecasting noise, as our control methods are not randomized.

The controller strategy is given by the incentive price Δp_t, which they post above the market. Therefore, given Δp_t the token owners' optimal strategy is to choose $\boldsymbol{\alpha_t}$ such that:

$$\begin{aligned} \underset{\boldsymbol{\alpha}_t}{\text{maximize}} \quad & U_t(\boldsymbol{\alpha}_t, \Delta p_t), \\ \text{subject to} \quad & \mathbf{1}^\top \boldsymbol{\alpha}_t = 1, \\ & \boldsymbol{\alpha}_t \geq \mathbf{0}. \end{aligned} \quad (8)$$

Moreover, by using Program (8), the controller can compute the token-owner's strategy for any incentive price Δp_t. This means that we can transition the control problem (6) into the setting with incentives by equating the number of tokens the controller buys back to the number of tokens the nodes agree to sell in each time step:

$$\frac{u_\tau^{\text{B}}}{p_\tau^{\text{Tok}} + \Delta p_\tau} = \alpha_\tau S_\tau, \qquad \forall \tau = t \cdots t+H, \quad (9)$$

where α_t values are computed as a function of incentive price through the program (8). This means that, in practice, the controller can account for selfish behavior to identify good incentive prices. We now formalize how the controller plays strategically by identifying the optimal incentive price Δp_t^*.

A. A Stackelberg Game for Strategic Pricing

Since the controller first posts a price Δp_t and the nodes respond with the fraction α_t of the holdings they wish to sell, we naturally have a leader-follower (Stackelberg) game. As mentioned above, we use (9) to constrain the tokens bought back by $\boldsymbol{\alpha}_t$. Then, recalling that the node's strategy is given by (8), the controller's optimization problem is:

$$\begin{aligned} \underset{u_{0:H-1}}{\text{minimize}} \quad & J(x_0, u_{0:H-1}), && (10) \\ \text{subject to} \quad & x_{t+1} = f(x_t, u_t, s_t), && \forall t \in \mathcal{T}, \\ & x_t \in \mathcal{X}, u_t \in \mathcal{U}, && \forall t \in \mathcal{T}, \\ & u_t^{\text{B}} = \alpha_t S_t(p_t^{\text{Tok}} + \Delta p_t), && \forall t \in \mathcal{T}, \\ & \boxed{\boldsymbol{\alpha}_t = \arg\max_{\boldsymbol{\alpha}_t} U_t(\boldsymbol{\alpha}_t, \Delta p_t),} \\ & \boxed{\mathbf{1}^\top \boldsymbol{\alpha}_t = 1, \quad \boldsymbol{\alpha}_t \geq \mathbf{0}.} \end{aligned}$$

Notice that the Stackelberg game can be cast as a *bi-level* optimization problem [34, 35], which is a nested optimization problem where an *outer* optimization problem involves a decision variable that is, in turn, the solution to a second *inner* problem. In our setting, the outer problem is the controller's non-convex control problem (6), which outputs Δp_t^* and requires $\boldsymbol{\alpha}_t^*$ as input. Here, $\boldsymbol{\alpha}_t^*$ is the solution to the inner utility maximization problem of the nodes. One observes that the inner utility maximization problem is convex and Slater's constraint qualification condition holds. Thus, the Karush-Kuhn-Tucker (KKT) conditions are sufficient for the optimality of the inner problem.

Since we have a Stackelberg Game, the horizon is finite, and a subgame perfect equilibrium can be found via backward induction [33]. First, the best response function of the nodes is calculated. Then, the controller picks an action maximizing its utility, anticipating the follower's best response. Then, it is clear that our method finds a subgame perfect equilibrium since the KKT conditions of the inner problem give a certificate of optimal play on the nodes part. Encoding them as extra constraints on the part of the controller simply gives an explicit route for backward induction in this game.

B. The Price of Incentives

When transitioning into a setting with strategic behavior, there will be a difference in network costs. Namely, the network cost in a perfect world, being where token owners willingly depart from their holdings at the controller's posted price, will be less than that incurred by a more robust controller that is accounting for strategic behavior. We quantify this as the ratio between the two costs, which we call the *price of incentives.* This is, suppose we are given an initial state x_0, sequence of forecasts $s_{0:H-1}$, and control methodology (e.g. MPC). Then, let $u_{0:H-1}$ be the controls selected by this methodology in the absence of strategic behavior, whereas $u_{0:H-1}^{U_t(\alpha_t,\Delta p_t)}$ are those selected in solving the Stackelberg game with token owner utility $U_t(\alpha_t, \Delta p_t)$. Then, the price of incentives for the instance $(x_0, s_{0:H-1}, U_t(\alpha_t, \Delta p_t))$ and the chosen control methodology is:

$$\frac{J(x_0, u_{0:H-1})}{J(x_0, u_{0:H-1}^{U_t(\alpha_t,\Delta p_t)})}. \tag{11}$$

Note that the price of incentives will largely depend on the risk aversion of the token owners, as parameterized by their discount factors. We present the above with respect to fixed methods and game instances, and leave a finer characterization relating discount factor values to the price of incentives for worst-case instances and arbitrary control methodologies for future work. That said, in our experiments, we compute the price of incentives incurred by our methods to understand the impact of strategic behavior on our controller, ControlPay.

VI. Experiments

The goal of our evaluation is to show that (i) using ControlPay enables us to achieve a more stable token price that approximates the target price trajectory, and (ii) we can reduce the network cost using our predictive controller instead of a reactive PID controller. We now describe our benchmark algorithms and evaluation metrics.

A. Evaluation Metrics and Benchmark Algorithms

We compare reserve controllers on the following metrics:

- **Stable Token Price**: The token price, volatility, and the mean squared error (MSE) from a reference price trajectory.
- **Network Cost Function**: A weighted sum of the tracking error (MSE between the token price and the reference price), and control effort.

We report these metrics for various realistic scenarios where (i) the supply of nodes out-paces the consumer demand, (ii) the supply and demand roughly match, and (iii) the supply lags the consumer demand. Our experiments compare the following schemes:

- **ControlPay**: Our proposed solution method, ControlPay, implements a predictive, optimal control scheme based on model predictive control (MPC) methods proposed in Section III.
- **Proportional Integral Derivative (PID)**: A traditional control method penalizing the deviation between current and desired states, with adjustments based on proportional, integral, and derivative terms.
- **No Control**: The worst case where the economy has no control and simply uses an income clearing strategy where the income, buybacks, and payments are equal.

B. Experiment Design

First, we evaluate our control system on synthetic node supply and consumer demand/income growth data. To create synthetic data, we start with a random starting point for demand/income D_0, Inc_0 and iteratively add increments drawn from the normal distributions $\mathcal{N}(0, 0.01 \cdot D_t^2)$ and $\mathcal{N}(0, 0.01 \cdot \text{Inc}_t^2)$ respectively. In all experiments, we have a noisy forecast $\hat{s}_t$ of demand and income growth for a horizon of H future steps. Following the experiments with the synthetic data, we extend our evaluation to real-world scenarios using demand data from the Helium DeWi network. Since Helium growth patterns are smooth, we use a classical Auto-Regressive Integrated Moving Average (ARIMA) forecasting model to predict network growth $H = 20$ days in advance.

We run 20 experiments-10 with synthetic data and 10 with real-world data with different target price trajectories. We use various patterns, covering from sigmoidal growth, where supply rapidly rises during the middle of network adoption but slowly tapers off, to logarithmic growth, where the demand steadily grows over time. In each experiment, the initial token price p_t^{Tok} is randomly chosen away from the reference and shows that the controller can dynamically regulate the system. All of our experiments are coded in Python, run in a few minutes on a standard laptop, and use the Gurobi Optimization package [36].

Does ControlPay reduce network cost?

We now evaluate our ultimate performance metric, the network cost, across a wide variety of growth patterns and initial conditions. Specifically, we used 3 growth patterns

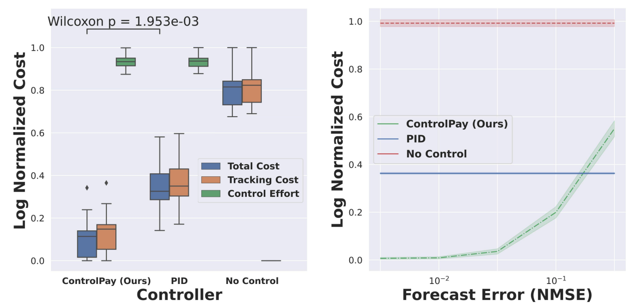

Fig. 3: **ControlPay outperforms the PID controller when forecast error is low.** (Left) Our ControlPay scheme achieves a lower total cost than the PID and the No Control benchmarks by exerting control effort to actively track the price trajectory. (Right) Both PID and ControlPay significantly outperform the No Control scheme. ControlPay significantly outperforms PID when the forecast error is low. As the estimation error increases, ControlPay starts to degrade in performance, whereas PID remains invariant. This difference is due to the predictive nature of ControlPay versus the reactive nature of PID.

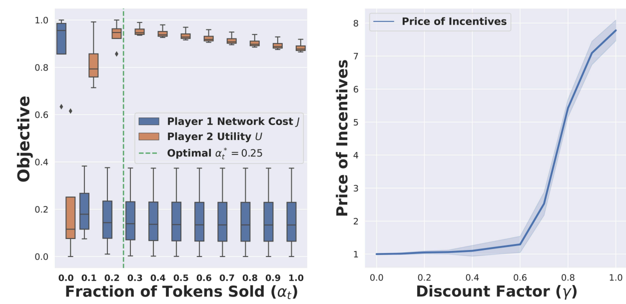

Fig. 4: **Price of Incentives.** (Left) Our bi-level optimization approach finds a game strategy to sell only α_t^* tokens (green line) to maximize node utility and minimize control cost compared to simply holding ($\alpha_t = 0$) or selling all tokens ($\alpha_t = 1$). (Right) We show how the price of incentives (Eq. 11) incurred by our controller varies over different values of the discount factor, γ. As γ increases, indicating that token owners are more reluctant to sell, the controller must offer a higher incentive price Δp_t to encourage token sales, thereby increasing the price of incentives.

with Gaussian noise (sigmoidal, logarithmic, exponential) and many scenarios where demand outstrips supply and vice versa. Fig. 3 shows the overall network cost, tracking error, and control effort for all 3 benchmarks. ControlPay achieves $2.4\times$ lower network cost than the PID heuristic. As shown in Fig. 3, the key reason for this difference is that PID is largely *reactive* – it proportionally responds to the current error and integrates the cumulative error but does not forecast the future system state accurately to optimize performance. In stark contrast, ControlPay explicitly solves an optimization problem to minimize the network cost.

How do ControlPay and PID compare given increasing forecast errors?

Fig. 3 (right) shows the total network costs for all 3 benchmarks across the Normalized Mean Squared Error (NMSE) of the forecast prediction. To show that, we add zero mean Gaussian noise to the forecast and plot the total cost for benchmarks. The plot shows that ControlPay achieves a lower total cost than the PID method when the forecast error is relatively small. As the estimation error increases, ControlPay begins to incur worse performance due to errors in forecasts propagating into future time steps. On the other hand, the performance of PID heuristics remains invariant across varying forecast errors due to its reactive nature. This exemplifies a tradeoff, being that ControlPay significantly outperforms the PID when forecasts are accurate. Therefore, our method reduces the problem of stabilizing a token economy to that of securing accurate forecasting data.

Does our Stackelberg game solution maximize each player's objective function?

Fig. 4 (left) illustrates the token-owning nodes' utility U_t (player 2) and network cost function J (player 1). We see that our bi-level optimization approach finds an optimal fraction $\alpha_t^* = 0.25$ of tokens to sell back, since the token owners' utility is maximized and network cost is minimized. Furthermore, the bi-level optimization outperforms naive strategies of holding all tokens with $\alpha_t = 0$ and selling back all tokens with $\alpha_t = 1$, in terms of maximizing the token owners' utility U_t. When our controller uses this equilibrium as a constraint (using Eq. 9), it becomes robust to strategic behavior arising from speculative behavior. As expected, the network cost J (blue boxplots) decreases as α_t increases since the controller can buy back more tokens than it wishes. Likewise, the node utility (orange) is low when they hold all their tokens ($\alpha_t = 0$) since the price drops because the controller can not remove tokens from circulation through buybacks.

What impact does strategic behavior have on network cost?

Fig. 4 (right) demonstrates the ratio between optimal network cost J and network cost with utility maximizer players, which is the price of incentives. We compare this ratio for different agent preferences by changing the discount factor γ. In effect, as the discount factor γ increases, the players become less risk-averse. For low to mid values of the discount factor γ, we see essentially no discrepancy between the network cost when strategic behavior is accounted for and ignored. However, it is not surprising to see an increase in the price of incentives once token owners increase their perceived value of possible future earnings. This is because, in this case, token owners are most hesitant to sell their holdings, and so more control effort must be exerted to track the price trajectory.

Note, prior studies use the overwhelming presence of mining pools in proof-of-work blockchains to argue that token owners are naturally risk-averse [37] in real-world systems. Hence, as the price of incentives remains low for low discount factor values corresponding to higher risk aversion, this suggests that adopting our Stackelberg game formulation in the real world would not incur noticeable extra network costs.

Does ControlPay yield a stable token price growth?

Fig. 5 shows two example trajectories of our system, where the left two columns are computed with real Helium data,

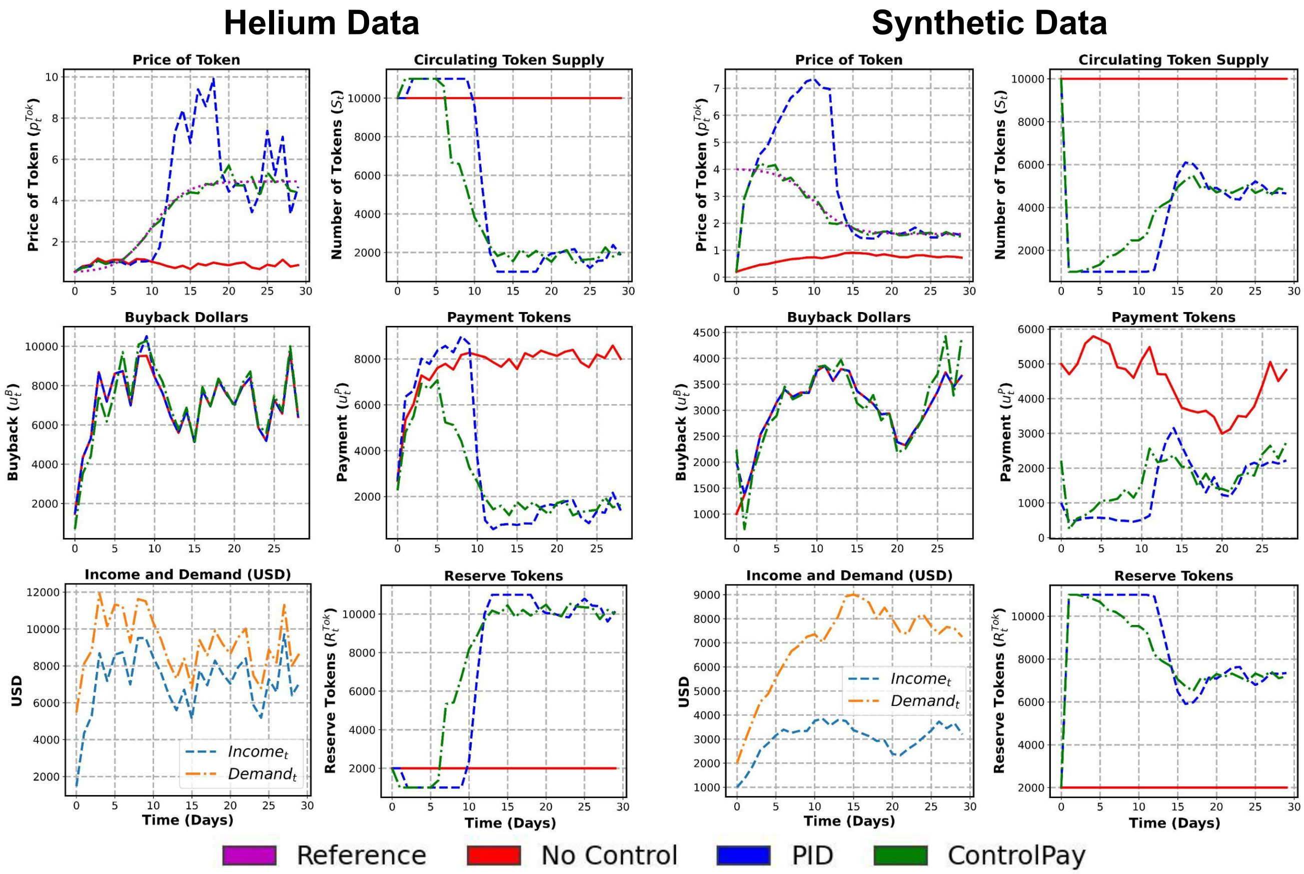

Fig. 5: **ControlPay outperforms benchmarks in simulations with both Helium and synthetic data.** The two left columns show trajectories using real Helium data, demonstrating that our ControlPay scheme (green) accurately follows the reference (purple), while reactive PID (blue) is highly oscillatory. The right columns validate our approach's adaptability using synthetic data, showing how ControlPay adjusts to a decreasing price trajectory. Unlike PID, ControlPay avoids overshooting, maintaining price stability even as demand shifts.

whereas the right two are with synthetic data. Our key result is found in the top left chart for the token price. Specifically, for the Helium data our ControlPay scheme (green) is able to track the reference (purple) extremely well, while the heuristic PID captures the general trend but is highly oscillatory since it is *reactive*. Crucially, the price plummets without control since the system pays too many tokens, causing inflation. However, our ControlPay scheme adaptively curtails token payments to reduce the circulating supply and avoid inflation (middle). Importantly, the income-clearing strategy given by No-Control (red) immediately pays out the exact same number of tokens it buys back from the market. Thus, the net change in the circulating token supply (and hence reserve) is zero, as indicated by the horizontal red lines for the token plots.

The right two columns confirm the generality of our approach – we can just as easily follow a smoothly decreasing price trajectory corresponding to a desire for a token price to become more affordable to incentivize increasing user adoption of a platform's services, for example. Crucially, the initial token price (column 3, row 1) is low and far from the reference for all schemes, but our ControlPay method (green) quickly rises to track the reference unlike PID (blue), which overshoots. Interestingly, we also see that the price without control is generally too low but shows an increasing trend over time since the income and demand governing this mechanism gradually increase over time. Finally, we see that ControlPay slowly increases adaptive payments after time step 11 to make the price decrease.

Limitations:

Our trace-driven simulations are limited by offline, historical Helium DeWi data. However, the growth of nodes and consumers might deviate from historical patterns if we actually implemented our proposed controller in the network. In future work, we plan to answer such "what-if" questions using recent advances in counterfactual analysis [38–40].

VII. Conclusion and Future Work

Our central thesis is that blockchain tokenomics should be programmable and dynamically adapt to changes in node growth and consumer demand. In particular, our key contribution is to model a token economy as a controlled dynamical system, which allows us to leverage model predictive control

(MPC) to design an adaptable payment controller, ControlPay, that meets the system architect's selection of a suitable network cost function. Additionally, we integrated a Stackelberg game approach to account for strategic behavior among token-owners, offering a control system that adjusts to disturbances caused by rational action-making on the part of the token-owners. Our empirical analysis underscores the superiority of our MPC based policy over the traditional PID heuristics and the efficacy of our Stackelberg game model in ensuring robustness against strategic token-owner actions.

We believe our work is timely as several blockchain projects are working with burn-and-mint strategies, and our framework enables us to improve those systems. In future work, we will work on integrating learning-based control methods to utilize the abundance of existing measurements and data to better adapt to changing market dynamics.

Acknowledgements

This work was supported by the National Science Foundation under grant no. 2148186 and 2133481 as well as federal agencies and industry partners as specified in the Resilient & Intelligent NextG Systems (RINGS) program. This article solely reflects the opinions and conclusions of its authors and does not represent the views of any sponsor.

References

[1] Amir Haleem, Andrew Allen, Andrew Thompson, Marc Nijdam, and Rahul Garg. Helium - a decentralized wireless network. https://whitepaper.io/document/649/helium-whitepaper, 2018.

[2] Anthony Levandowski. Intro to pollen mobile. https://docs.pollenmobile.io/pollen-mobile-docs/white-paper, 2021.

[3] Yuma Rao, Jacob Steeves, Ala Shaabana, Daniel Attevelt, and Matthew McAteer. Bittensor: A peer-to-peer intelligence market, 2021.

[4] Protocol Labs. Filecoin: A decentralized storage network. https://whitepaper.io/document/599/filecoin-whitepaper, 2020.

[5] Storj Labs. Storj: A decentralized cloud storage network framework. https://www.storj.io/storjv3.pdf, 2018.

[6] ComputeCoin Network. Computecoin network: The infrastructure of web 3.0 and the metaverse. https://whitepaper.io/document/649/helium-whitepaper, 2020.

[7] Burn and mint equilibrium. https://messari.io/report/burn-and-mint-equilibrium, 2019. [Online; accessed 19-October-2022].

[8] Grady Williams, Paul Drews, Brian Goldfain, James M Rehg, and Evangelos A Theodorou. Information-theoretic model predictive control: Theory and applications to autonomous driving. IEEE Transactions on Robotics, 34(6):1603–1622, 2018.

[9] Mohak Bhardwaj, Balakumar Sundaralingam, Arsalan Mousavian, Nathan D Ratliff, Dieter Fox, Fabio Ramos, and Byron Boots. Storm: An integrated framework for fast joint-space model-predictive control for reactive manipulation. In Conference on Robot Learning, pages 750–759. PMLR, 2022.

[10] Oguzhan Akcin, Po han Li, Shubhankar Agarwal, and Sandeep P. Chinchali. Decentralized data collection for robotic fleet learning: A game-theoretic approach. In 6th Annual Conference on Robot Learning, 2022. URL https://openreview.net/forum?id=iM932PeUi_7.

[11] Akshay Agrawal, Shane Barratt, Stephen Boyd, and Bartolomeo Stellato. Learning convex optimization control policies. In Learning for Dynamics and Control, pages 361–373. PMLR, 2020.

[12] Michael Zargham, Zixuan Zhang, and Victor Preciado. A state-space modeling framework for engineering blockchain-enabled economic systems. arXiv preprint arXiv:1807.00955, 2018.

[13] Zixuan Zhang, Michael Zargham, and Victor M Preciado. On modeling blockchain-enabled economic networks as stochastic dynamical systems. Applied Network Science, 5(1):1–24, 2020.

[14] Carey Caginal. A dynamical systems approach to cryptocurrency stability. arXiv preprint arXiv:1805.03143, 2018.

[15] Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system. Decentralized Business Review, page 21260, 2008.

[16] Ittay Eyal and Emin Gün Sirer. Majority is not enough: Bitcoin mining is vulnerable. Communications of the ACM, 61(7):95–102, 2018.

[17] Miles Carlsten, Harry Kalodner, S Matthew Weinberg, and Arvind Narayanan. On the instability of bitcoin without the block reward. In Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, pages 154–167, 2016.

[18] Ittay Eyal. The miner's dilemma. In 2015 IEEE Symposium on Security and Privacy, pages 89–103. IEEE, 2015.

[19] Tim Roughgarden. Transaction fee mechanism design. ACM SIGecom Exchanges, 19(1):52–55, 2021.

[20] Hao Chung and Elaine Shi. Foundations of transaction fee mechanism design. arXiv preprint arXiv:2111.03151, 2021.

[21] Andrea Bacchiocchi and Germana Giombini. An optimal control problem of monetary policy. Discrete and Continuous Dynamical Systems - B, 26(11):5769–5786, 2021. ISSN 1531-3492. doi: 10.3934/dcdsb.2021224. URL /article/id/04be7198-482b-4b01-92a1-3f76f3d0d6b2.

[22] Flint Brayton, Thomas Laubach, and David L. Reifschneider. Optimal-Control Monetary Policy in the FRB/US Model. FEDS Notes 2014-11-21-1, Board of Governors of the Federal Reserve System (U.S.), November 2014. URL https://ideas.repec.org/p/fip/fedgfn/2014-11-21-1.html.

[23] David Shepherd, Rebeca I. Muñoz Torres, and George Saridakis. Monetary policy rules with pid control features: evidence from the uk, usa and eu. International Review of Applied Economics, 33(6): 737–755, 2019. doi: 10.1080/02692171.2019.1585766. URL https://doi.org/10.1080/02692171.2019.1585766.

[24] Raymond Hawkins, Jeffrey Speakes, and Dan Hamilton. Monetary policy and pid control. Journal of Economic Interaction and Coordination, 10(1):183–197, 2015. URL https://EconPapers.repec.org/RePEc:spr:jeicoo:v:10:y:2015:i:1:p:183-197.

[25] Paul Snow, Brian Deery, Jack Lu, David Johnston, and Peter Kirby. Factom: Business processes secured by immutable audit trails on the blockchain. Whitepaper, Factom, November, 111, 2014.

[26] Summoning the money god: Applying control theory to cryptocurrency monetary policy. https://medium.com/reflexer-labs/summoning-the-money-god-2a3f3564a5f2, 2021. [Online; accessed 10-June-2021].

[27] Evan Kereiakes, Marco Di Maggio Do Kwon, and Nicholas Platias. Terra money: Stability and adoption. White Paper, Apr, 2019.

[28] Vincent Messié, Gaël Fromentoux, Xavier Marjou, and Nathalie Labidurie Omnes. Baladin for blockchain-based 5g networks. In 2019 22nd Conference on Innovation in Clouds, Internet and Networks and Workshops (ICIN), pages 201–205. IEEE, 2019.

[29] Doug Petkanics et al. Protocol and economic incentives for a decentralized live video streaming network, 2018.

[30] Francesco Borrelli, Alberto Bemporad, and Manfred Morari. Predictive control for linear and hybrid systems. Cambridge University Press, 2017.

[31] Riccardo Bonalli, Abhishek Cauligi, Andrew Bylard, and Marco Pavone. Gusto: Guaranteed sequential trajectory optimization via sequential convex programming. In 2019 International conference on robotics and automation (ICRA), pages 6741–6747. IEEE, 2019.

[32] Danylo Malyuta, Taylor P Reynolds, Michael Szmuk, Thomas Lew, Riccardo Bonalli, Marco Pavone, and Behcet Acikmese. Convex optimization for trajectory generation. arXiv preprint arXiv:2106.09125, 2021.

[33] Martin J Osborne et al. An introduction to game theory, volume 3. Oxford university press New York, 2004.

[34] Benoît Colson, Patrice Marcotte, and Gilles Savard. An overview of bilevel optimization. Annals of operations research, 153(1):235–256, 2007.

[35] Jonathan F Bard. Practical bilevel optimization: algorithms and applications, volume 30. Springer Science & Business Media, 2013.

[36] LLC Gurobi Optimization. Gurobi optimizer reference manual, 2018.

[37] Xi Chen, Christos Papadimitriou, and Tim Roughgarden. An axiomatic approach to block rewards. In Proceedings of the 1st ACM Conference on Advances in Financial Technologies, pages 124–131, 2019.

[38] Stephen L Morgan and Christopher Winship. Counterfactuals and causal inference. Cambridge University Press, 2015.

[39] Judea Pearl. Causal inference. Causality: objectives and assessment, pages 39–58, 2010.

[40] Anish Agarwal, Devavrat Shah, and Dennis Shen. Synthetic interventions. arXiv preprint arXiv:2006.07691, 2020.